

KAN-ResCNN: An Advanced Deep Learning Model for Distinguishing Z Boson Decay Modes*

Wufeng Liu,^{1,†} Feihu Wang,¹ Pengsong Jiang,¹ Renjie Wei,¹ Zhouli Zhang,² Yuhong Yu,² Xiangman Liu,² Yan Li,¹ and Longfei Li¹

¹*Henan University of Technology, Zhengzhou 450001, China*

²*Institute of modern physics, Chinese Academy of Sciences, Lanzhou 730000, China*

Addressing the growing demand for intelligent automation in particle physics analysis, this study introduces an innovative deep learning framework for classifying Z boson leptonic decay channels ($Z \rightarrow \mu^+ \mu^-$ vs. $Z \rightarrow e^+ e^-$). We propose KAN-ResCNN, a hybrid architecture synergistically integrating three advanced paradigms: Kolmogorov-Arnold Networks, Residual Networks, and Convolutional Neural Networks. The model employs a dual-branch feature extraction system where the Residual Networks backbone captures hierarchical physical patterns through residual learning while the Convolutional Neural Networks stream processes localized feature correlations. Theoretical physics principles are systematically encoded through Kolmogorov-Arnold Networks driven nonlinear classification, ensuring physically interpretable decision boundaries. Evaluated on a publicly available experimental dataset from CERN's collider experiments, the proposed model achieves state-of-the-art performance with 94.73% classification accuracy and 94.75% F1-score, demonstrating statistically significant improvements over conventional machine learning baselines and monolithic deep architectures. This work establishes a novel paradigm for high-energy physics data analysis that simultaneously optimizes discriminative power and physical consistency. The architecture's modular design permits direct extension to other quantum chromodynamics processes, providing critical technical infrastructure for next-generation intelligent particle physics analysis systems. Key implementation details and ablation studies are presented to elucidate the relative contributions of each architectural component.

Keywords: Z boson decay, Kolmogorov-Arnold Network, Residual Network, Deep Learning

I. INTRODUCTION

Z bosons, as mediators of the weak interaction in the Standard Model, hold irreplaceable value in particle physics research [1]. The lepton decay modes of the Z boson, $Z \rightarrow \mu^+ \mu^-$ (Zmumu) and $Z \rightarrow e^+ e^-$ (Zee), provide a critical window for exploring the fundamental particle interaction mechanisms [2]. These processes are not only central to testing the precision of the Standard Model [3] but may also offer clues to new physical phenomena, such as new particles or interaction mechanisms beyond the Standard Model [4]. The challenge of distinguishing similar decay modes, such as Zmumu and Zee, remains a bottleneck in experimental analysis. As the data produced by large-scale facilities like the Large Hadron Collider (LHC) grows exponentially [5], traditional analysis methods face significant challenges in handling high-dimensional, nonlinear data and fully exploiting potential physical insights. Therefore, the development of an automated classification tool that combines high precision with physical interpretability has become an urgent need in particle physics research [6]. Such tools could not only enhance the efficiency of Standard Model verification but may also provide crucial data support for the discovery of new physical phenomena through precise identification of decay modes.

In this study, we use the Z boson dataset from CERN open data repository, which is primarily derived from the datasets

contributed by McCauley and Thomas. These datasets are based on the Run2011A primary datasets, including SingleElectron, SingleMu, DoubleElectron, and DoubleMu (<https://opendata.cern.ch/record/545>). Using this dataset, we propose a novel classification model, KAN-ResCNN, which integrates deep learning with prior physical knowledge [7]. This model innovatively combines the Kolmogorov-Arnold Network (KAN) [8], Residual Network (ResNet) [9], and Convolutional Neural Network (CNN) [10], aiming to address the efficiency and interpretability shortcomings of traditional methods in classifying Z boson decay modes [11]. Specifically, KAN-ResCNN employs a dual-branch architecture: the ResNet branch extracts deep physical features via residual structures, the CNN branch captures local feature correlations, and the KAN module applies a theory-driven nonlinear mapping for classification decisions. This architecture not only enhances the model's ability to represent complex physical laws but also provides physical interpretability through the visualization capability of KAN [12].

Experimental results show that, on the CERN Open Dataset, KAN-ResCNN achieves an accuracy of 94.73% and an F1 score of 94.75% in the binary classification task of Zmumu versus Zee, significantly outperforming traditional models and other classification models for Z boson decay modes [13]. The contributions of this study can be summarized as follows: (1) The first particle physics data analysis model integrating KAN with a deep residual-convolutional architecture is proposed; (2) A dual-branch design is employed to balance physical feature extraction with data-driven learning; (3) A synergistic improvement in classification accuracy and interpretability is achieved. This work provides an efficient and reliable analytical tool for high-energy physics experiments and offers methodological support for the intelligent transformation of particle physics research. Future work

* We would like to thank McCauley and Thomas; We also thank CERN for providing experimental data on the CERN open data platform. This work was supported by Natural Science Foundation of Henan Province under Grant (232300420114)

† Corresponding author, lwf@haut.edu.cn

61 will extend this model to other particle decay modes, such as
 62 Higgs boson two-photon decays, for identification tasks [14].

63 II. RELATED WORK

64 A. The Evolution of Residual Networks in Particle Physics

65 Residual Networks (ResNet) have revolutionized deep
 66 learning through their intrinsic skip-connection mechanism,
 67 which effectively mitigates gradient degradation in deep ar-
 68 chitectures while enabling stable training of networks with
 69 more than 100 layers [15]. This paradigm has recently
 70 permeated high-energy physics (HEP) research, particu-
 71 larly for analyzing detector-generated one-dimensional fea-
 72 ture vectors. Pioneering applications include Ref. [16],
 73 which implemented transfer learning through fine-tuning of
 74 ResNet-50 architectures on LHC-generated 2D histograms
 75 (signal+background vs. background-only configurations),
 76 demonstrating 90.2% classification accuracy for dark mat-
 77 ter signatures at signal-to-background ratios as low as 0.008.
 78 Concurrently, Ref. [17] developed a dual-ResNet framework
 79 for plasma jet dynamics prediction, leveraging computational
 80 fluid dynamics (CFD) simulations of multi-arc plasma spray-
 81 ing systems. Their cascaded architecture achieved relative er-
 82 rors below 3.5% in particle velocity/temperature predictions,
 83 reducing computational costs by two orders of magnitude
 84 compared to conventional CFD solvers—a critical advance-
 85 ment for real-time plasma diagnostics. While promising, cur-
 86 rent ResNet adaptations in High Energy Physics (HEP) reveal
 87 several limitations: they primarily rely on pretrained vision
 88 models instead of physics-optimized architectures, underuti-
 89 lize domain knowledge in network design, and lack explicit
 90 mechanisms for enforcing theoretical constraints.

91 To address these issues, we have developed a modi-
 92 fied ResNet architecture that employs 1D convolutional ker-
 93 nels specifically optimized for collider detector outputs [18].
 94 Through a parallel design with a CNN branch, the model
 95 extracts both deep physical laws and local feature corre-
 96 lations [19]. Additionally, the introduction of the KAN module
 97 provides a theory-driven nonlinear mapping, compensating
 98 for the traditional ResNet’s limitations in embedding physi-
 99 cal prior knowledge [20]. As shown in Fig. 1, the input data
 100 first passes through the input layer, where local features are
 101 extracted via a convolutional layer. Subsequently, the data un-
 102 dergoes normalization and nonlinear transformation through
 103 a batch normalization layer followed by a ReLU activation
 104 function [21]. The data then sequentially passes through four
 105 residual blocks (illustrated in Fig. 2), with each block consist-
 106 ing of convolutional layers, batch normalization layers, ReLU
 107 activations, and skip connections, progressively extracting
 108 deeper-level features. Finally, by a flattening layer that trans-
 109 forms the multi-dimensional data into a one-dimensional vec-
 110 tor, preparing it for the subsequent fully connected layers or
 111 output layer [22]. This architecture effectively addresses the
 112 vanishing gradient problem by improving gradient flow dur-
 113 ing backpropagation, which allows for the successful training
 114 of very deep models, especially in tasks like image classifica-

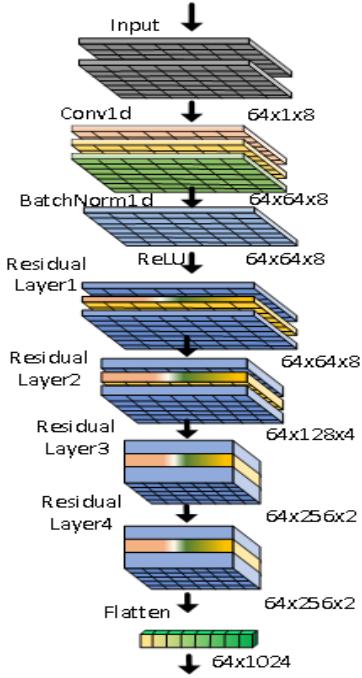


Fig. 1. Resnet Architecture Diagram.

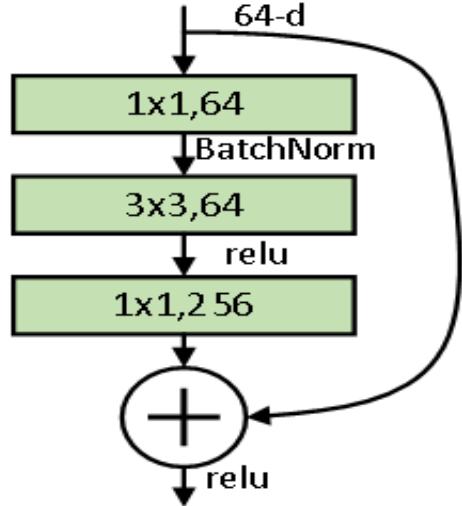


Fig. 2. Residual Block Architecture Diagram.

117 tion and object recognition tasks [23].

118 B. Local Feature Extraction of Convolutional Neural 119 Networks

120 Convolutional Neural Networks (CNNs) have demon-
 121 strated significant advantages in processing raw data in
 122 particle physics due to their local perception and pa-
 123 rameter sharing characteristics [24]. By stacking convolu-
 124 tional and pooling layers to extract spatial local fea-

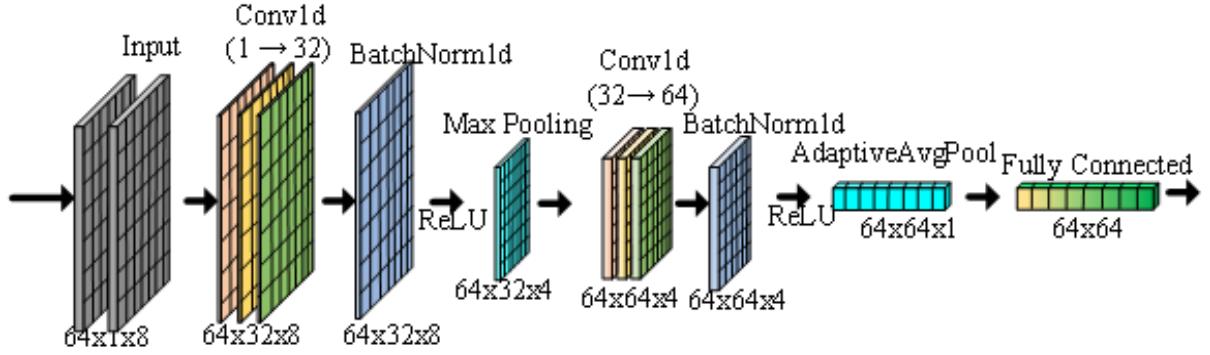


Fig. 3. Convolutional Neural Network Architecture Diagram.

125 tures, CNNs have achieved remarkable success in tasks
 126 such as jet classification [25] and event reconstruction [26].
 127 However, traditional CNNs still have two major limita-
 128 tions in particle physics analysis: first, the local percep-
 129 tion property of CNNs causes the model to overly focus
 130 on spatial patterns (such as track topology) while neglect-
 131 ing global physical laws (such as statistical properties of
 132 the lepton pair invariant mass); second, the "black-box" na-
 133 ture of CNNs makes their decision-making process diffi-
 134 cult to associate with Standard Model theory, limiting the
 135 interpretability of scientific discoveries [27]. To address
 136 these issues, this study introduces three improvements to
 137 the CNN architecture: first, replacing traditional 2D con-
 138 volutions with 1D convolutions, which allows direct pro-
 139 cessing of one-dimensional time-series features (such as
 140 track curvature and energy deposition distributions) produced
 141 by detectors, thereby avoiding information loss caused
 142 by the transformation of 2D images [28]; second, using
 143 adaptive average pooling to map input features of varying
 144 lengths to fixed-dimensional outputs, ensuring compatibility
 145 with subsequent modules [29]; and third, designing paral-
 146 lel CNN and ResNet branches, where the CNN branch fo-
 147 cuses on local pattern extraction (such as track topology and
 148 cluster shape), while the ResNet branch captures deep fea-
 149 tures through residual connections. Ultimately, the features
 150 are complemented through tensor concatenation, effectively
 151 balancing local perception with global pattern modeling.

152 The convolutional neural network architecture employed
 153 in this study is shown in Fig. 3. The network begins with
 154 an input layer, which is responsible for receiving the exten-
 155 sional data. This is followed by a convolutional layer with 32
 156 filters, designed to extract local features from the input data.
 157 The ReLU activation function is applied to introduce non-
 158 linearity into the model. Batch normalization is then per-
 159 formed to standardize the mean and standard deviation of
 160 each feature channel, enhancing the stability of the model.
 161 A max-pooling layer is employed to reduce the dimen-
 162 sionality while preserving the most significant features. Another
 163 convolutional layer follows, utilizing 64 filters to extract more
 164 complex and higher-level features, with ReLU activation and
 165 batch normalization applied to the output. Subsequently, an
 166 adaptive average pooling layer is used to capture global fea-

167 tures, providing a fixed-size output. Finally, a fully connected
 168 layer maps the extracted features to the final output space, fa-
 169 cilitating the decision-making process.

C. Design of the Kolmogorov-Arnold Network Module

170 The Kolmogorov-Arnold Network (KAN) is a neural net-
 171 work architecture based on the Kolmogorov-Arnold rep-
 172 resentation theorem [8], designed to approximate high-
 173 dimensional functions through piecewise polynomial approx-
 174 imations while maintaining both model interpretability and
 175 computational efficiency [30]. Traditional neural networks,
 176 such as Multi-Layer Perceptrons (MLPs), although possess-
 177 ing powerful function approximation capabilities, suffer from
 178 the "black-box" nature [31], making their decision-making
 179 process difficult to correlate with physical laws, which limits
 180 their applicability in scientific discovery. KAN enhances in-
 181 terpretability significantly by incorporating learnable spline
 182 functions and basis functions, while preserving computa-
 183 tional efficiency. The basic structure of the KAN network,
 184 as depicted in Fig. 4, comprises multiple hierarchical layers
 185 through which data flows. In each layer, the input data un-
 186 dergoes processing by a specific set of functions, which then
 187 transmits the processed output to the subsequent layer. The
 188 activation function used at each node is illustrated, with the
 189 B-spline function being employed in this network. The role of
 190 the B-spline function is critical, as its graphical representation
 191 demonstrates how the "grid extension technique" facilitates
 192 the transition between coarse and fine grids. This method,
 193 in conjunction with the dynamic adjustment of the network's
 194 multi-layered structure and activation functions, enables the
 195 network to effectively handle high-dimensional data. Such a
 196 design not only accommodates varying data resolutions but
 197 also optimizes overall performance by fine-tuning the preci-
 198 sion of the activation functions.

1. Core Design of KAN

200 The Kolmogorov-Arnold representation theorem forms the
 201 theoretical foundation of KAN. This theorem asserts that

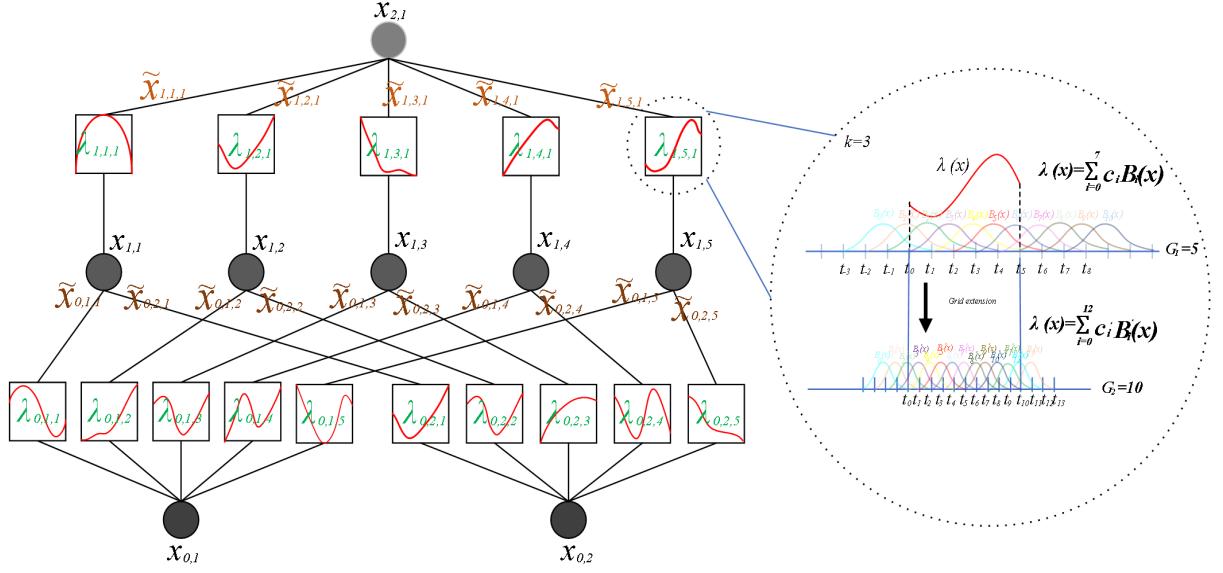


Fig. 4. The hierarchical architecture of the KAN network and its activation functions.

any multivariate continuous function $f(x_1, x_2, \dots, x_n)$ can be represented as a sum of a finite number of univariate functions:

$$f(x_1, x_2, \dots, x_n) = \sum_{q=1}^{2n+1} \Phi_q \left[\sum_{p=1}^n \lambda_{q,p}(x_p) \right]. \quad (1)$$

In this context, Φ_q is the outer function responsible for combining the outputs of lower-dimensional functions into the final result, while $\lambda_{q,p}$ represents the inner functions that handle each input dimension individually. Based on this, KAN decomposes the high-dimensional function into a combination of multiple lower-dimensional functions, achieved through the following key components:

Piecewise Polynomial Approximation: KAN utilizes B-spline basis functions to perform piecewise polynomial approximation on the input data. B-spline basis functions have local support and smoothness, enabling them to effectively capture the local features of the input data. The mathematical expression for a B-spline basis function is:

$$B_{i,k}(x) = \frac{x - t_i}{t_{i+k} - t_i} B_{i,k-1}(x) + \frac{t_{i+k+1} - x}{t_{i+k+1} - t_{i+1}} B_{i+1,k-1}(x), \quad (2)$$

where $B_{i,k}(x)$ is the i -th B-spline basis function of degree k , and t_i is the knot vector that defines the support interval of the spline function. This expression is recursively defined, with lower-degree spline functions (of degree $k-1$) being linearly combined to generate higher-degree spline functions (of degree k).

Fusion of Basis Functions and Spline Functions: KAN combines basis functions and spline functions to capture both the global trends and local details of the input data. The basis functions compute global features through linear transformations, while the spline functions calculate local features via

B-spline basis functions and learnable weights. This design allows KAN to model both global patterns and local structures simultaneously. The output can be expressed as:

$$y = \text{BaseOutput}(x) + \text{SplineOutput}(x), \quad (3)$$

$$\text{BaseOutput}(x) = \text{SiLU}(W_{\text{base}} \cdot x), \quad (4)$$

$$\text{SplineOutput}(x) = \sum_i w_i \cdot B_i(x). \quad (5)$$

The BaseOutput branch of the basis function computes global features by applying a linear transformation (W_{base}) followed by the SiLU activation function, thereby capturing the overall trend of the input data. The SplineOutput branch of the spline function computes local features by utilizing B-spline basis functions $B_i(x)$ and learnable weights w_i , effectively capturing the detailed patterns within the input data.

Dynamic Grid Update: To enhance the model's adaptability to the distribution of input data, KAN introduces a dynamic grid update mechanism. This mechanism adjusts the grid points of the spline function dynamically based on the distribution of the input data, ensuring the model can adaptively capture the key features of the data. The grid update formula is:

$$\text{grid} = \text{grid}_{\text{uniform}} \cdot \tau + \text{grid}_{\text{adaptive}} \cdot (1 - \tau), \quad (6)$$

where $\text{grid}_{\text{uniform}}$ represents a uniform grid, which is evenly distributed based on the range of the input data, and $\text{grid}_{\text{adaptive}}$ is an adaptive grid, which dynamically adjusts according to the actual distribution of the input data. τ is the grid adjustment parameter that controls the weighting between the uniform and adaptive grids.

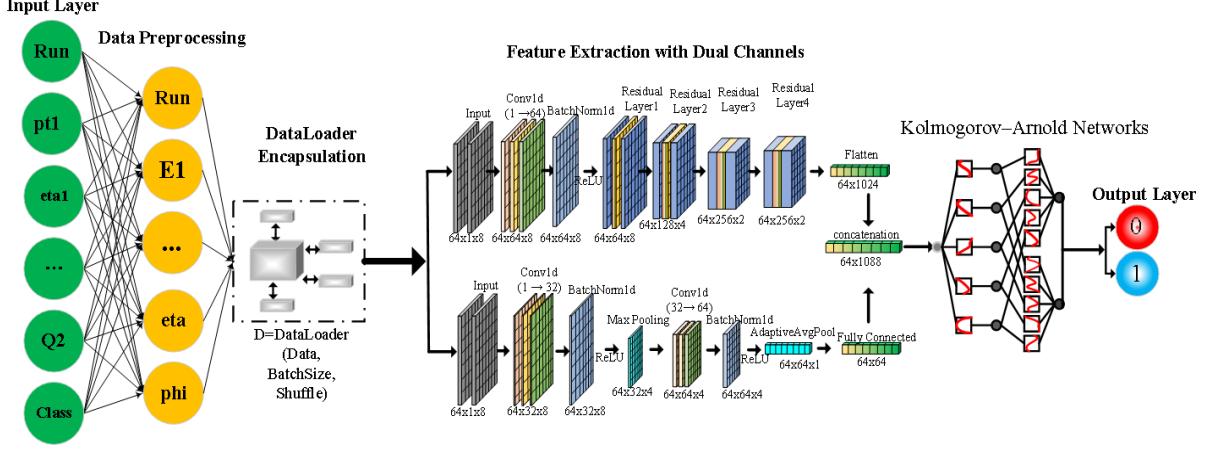


Fig. 5. Overall Structure Diagram.

262

2. Innovations in KAN

263 Traditional implementations require the expansion of all
 264 intermediate variables to compute different activation func-
 265 tions, leading to significant memory overhead. KAN reduces
 266 memory costs significantly by reformulating the computa-
 267 tion process, transforming the activation function computa-
 268 tion into a linear combination of basis functions. The original
 269 KAN implementation used L1 regularization on intermediate
 270 tensors for nonlinear operations, which is incompatible with
 271 the reformulated computation. In this revised approach, the
 272 L1 regularization is applied to the weights, and the formula is
 273 as follows:

$$L_{\text{reg}} = \lambda_1 \sum |W_{\text{spline}}| + \lambda_2 \sum p \log p. \quad (7)$$

274 Activation Term: $\sum |W_{\text{spline}}|$ represents the L1 regularization
 275 applied to the spline function weights, which serves to con-
 276 strain the model's complexity and prevent overfitting. En-
 277 tropy Term: $\sum p \log p$ is the entropy regularization applied to
 278 the spline function weights, designed to enhance the sparsity
 279 and interpretability of the model. λ_1 and λ_2 are the regular-
 280 ization weights for the activation and entropy terms, respec-
 281 tively. The regularization loss function improves the general-
 282 ization ability and interpretability of the model by constrain-
 283 ing the distribution of the spline function weights.

285

III. METHOD

286

A. Overall Architecture Design

287 The proposed KAN-ResCNN model employs a dual-
 288 branch parallel architecture aimed at simultaneously captur-
 289 ing local patterns at the detector level and global physical
 290 laws. By integrating the deep feature extraction capabil-
 291 ity of Residual Networks (ResNet), the local pattern capture
 292 ability of Convolutional Neural Networks (CNN), and the
 293 interpretable mapping of the Kolmogorov–Arnold Network

294 (KAN), this model provides a comprehensive approach. As
 295 shown in Fig. 5, the input to the model is a one-dimensional
 296 feature vector representing the Z boson decay event, from
 297 which classification decisions are made through the following
 298 steps. The CNN branch extracts local physical features using
 299 1D convolutional layers and adaptive average pooling, while
 300 the ResNet branch captures deep global features through the
 301 stacking of residual blocks. After the concatenation of fea-
 302 tures from both branches, they are fed into the KAN classifier,
 303 which leverages a combination of spline functions and basis
 304 functions to achieve a theory-driven nonlinear mapping.

305 B. Key Technological Innovations and Implementation Details

306 1. Dynamic Dimension Fusion and Adaptive Classification

307 During the initialization phase, the model achieves flexible
 308 feature fusion by dynamically calculating the feature dimen-
 309 sions. The input data undergoes forward propagation, auto-
 310 matically computing the concatenated dimensions of the out-
 311 puts from both branches. First, a random tensor representing
 312 the input is passed through the CNN branch to calculate the
 313 feature dimensions of that branch. Then, the ResNet branch
 314 processes the input data and computes the corresponding fea-
 315 ture output. Ultimately, the total dimension of the feature
 316 fusion is obtained by summing the feature dimensions from
 317 both branches. This dynamic dimension computation mecha-
 318 nism enables the model to adapt to variations in input feature
 319 lengths, ensuring the robustness of feature concatenation. Fi-
 320 nally, the fused features are processed by the KAN classi-
 321 fier, which performs theory-driven nonlinear mapping on the
 322 combined information. The mathematical formulation is as
 323 follows:

$$y = \text{KAN}(\text{Concat}(F_{\text{CNN}}, F_{\text{ResNet}})), \quad (8)$$

324 where F_{CNN} and F_{ResNet} represent the feature vectors of the
 325 CNN and ResNet branches, respectively.

327

2. Residual Block Optimization Design

328 In response to the characteristics of particle physics data,
 329 this study improves upon the traditional residual block struc-
 330 ture. Firstly, the conventional two-dimensional convolutional
 331 residual block is replaced with a one-dimensional convolu-
 332 tion, aiming to better align with the temporal signal character-
 333 istics of the detector. Secondly, to mitigate the common issue
 334 of vanishing gradients in deep networks, a shortcut connec-
 335 tion is introduced within the residual link, thereby improving
 336 gradient stability. Additionally, a channel expansion strategy
 337 is employed, progressively increasing the number of channels
 338 at each layer, effectively enhancing the feature representation
 339 capacity. The forward propagation process of the residual
 340 block can be formally expressed as:

$$341 \quad F(x) = \text{ReLU}(f_2(f_1(x)) + S(x)), \quad (9)$$

342 where f_1 and f_2 represent convolutional layers, and $S(x)$ de-
 343 notes the shortcut connection branch. This modified residual
 344 block structure significantly enhances the model's ability to
 345 process particle physics data.

346 3. Physics-Inspired Feature Extraction

347 In response to the characteristics of detector data, an op-
 348 timized feature extraction module is designed in this study,
 349 comprising both CNN and ResNet branches. In the CNN
 350 branch, two-stage 1D convolutions are employed to extract
 351 track topology features, capturing local patterns. Subse-
 352 quently, global average pooling (AdaptiveAvgPool1d) is ap-
 353 plied to compress the dimensionality of the features, generat-
 354 ing fixed-dimensional feature representations. During model
 355 initialization, the output dimension of the CNN branch is
 356 computed using synthetic input data to accommodate varying
 357 feature lengths of the input. The ResNet branch, on the other
 358 hand, learns global physical patterns progressively by stack-
 359 ing four residual blocks. This design enables the model to
 360 fully leverage physics-inspired principles to uncover deeper
 361 patterns within the detector data.

362 IV. EXPERIMENTS

363 A. Experimental Setup and Dataset Preprocessing

364 The experimental dataset employed in this study orig-
 365 inates from the CMS experiment at the CERN-LHC,
 366 specifically derived from the " Datasets derived from the
 367 Run2011A SingleElectron, SingleMu, DoubleElectron, and
 368 DoubleMu primary datasets " curated by McCauley and
 369 Thomas. Our analysis focuses on the critical classification
 370 task of distinguishing between two key weak interaction de-
 371 cay channels of the Z boson: Zmumu and Zee. These chan-
 372 nels exhibit pronounced physical similarities, presenting two
 373 primary classification challenges: (i) topological consistency

374 in final-state particle signatures, and (ii) overlapping lepton-
 375 pair invariant mass distributions within the 60–120 GeV/c²
 376 range. The curated dataset comprises 20,000 high-confidence
 377 collision events, each systematically annotated with 10 fun-
 378 damental physical observables: Run (the run number of the
 379 event), Event (the event number), pt1 and pt2 (the transverse
 380 momentum of the lepton, either a muon or an electron, in
 381 GeV), Q1 and Q2 (the charge of the lepton, either a muon
 382 or an electron), phi1 and phi2 (the phi angle of the lepton,
 383 either a muon or an electron, in radians), eta1 and eta2 (the
 384 pseudorapidity of the lepton, either a muon or an electron).
 385 These parameters collectively encode the kinematic charac-
 386 teristics of decay products and electromagnetic interaction
 387 properties. The intrinsic correlations among parameters, cou-
 388 pled with sub-detector-level measurement precision, estab-
 389 lish a rigorous benchmark for evaluating machine learning
 390 classifiers in high-energy physics applications. Furthermore,
 391 the dataset undergoes standardized preprocessing to ensure
 392 cross-experimental consistency in physical parameterization.
 393 This meticulously processed dataset serves as a challenging
 394 testbed for developing advanced classification algorithms ca-
 395 pable of addressing nuanced pattern recognition tasks in par-
 396 ticle physics phenomenology.

397 The dataset was initially imported via the Pandas library in
 398 Python, followed by fundamental preprocessing operations.
 399 The categorical labels ("Zmumu" and "Zee") were numer-
 400 ically encoded (0 and 1) to support deep learning applica-
 401 tions. The data preprocessing involved two key stages: di-
 402 mensionality reduction via feature selection, along with out-
 403 lier detection and normalization. Outlier analysis employed
 404 boxplot visualization (Fig. 6), where vertical axes display
 405 normalized feature distributions (horizontal range 0-1). Pro-
 406 longed box lengths indicated dispersed distributions with po-
 407 tential outliers, defined mathematically as observations be-
 408 yond Q1-1.5IQR or Q3+1.5IQR boundaries (IQR = interquar-
 409 tile range). Critical examination revealed significant outlier
 410 presence in pt1, pt2, Q1, and Q2 features. Empirical val-
 411 idation demonstrated substantial experimental impact from
 412 Q1/Q2 outliers, prompting their removal to enhance data reli-
 413 ability. For feature optimization, a correlation matrix visual-
 414 ization (Fig. 7) was employed, where axes denote dataset fea-
 415 tures and chromatic intensity reflects Pearson correlation co-
 416 efficients (-1 to 1). Features exhibiting absolute correlations
 417 exceeding 0.7 were identified as redundant, with strategic re-
 418 tention of single representatives from collinear groups. This
 419 approach simultaneously enabled identification of predictive
 420 features through their correlation patterns with key variables,
 421 while the matrix's inherent symmetry validated dataset con-
 422 sistency. Post-feature elimination, MinMaxScaler standar-
 423 dization was implemented using the transformation:

$$424 \quad X_{\text{norm}} = \frac{(X - \mu)}{\sigma}, \quad (10)$$

425 where μ and σ represent feature-specific mean and stan-
 426 dard deviation, respectively, mapping values to [0,1] inter-
 427 vals. This preprocessing pipeline established a robust foun-
 428 dation for subsequent model development by ensuring di-
 429 mensional consistency, eliminating noise, and maintaining



Fig. 6. The box plot generated from the dataset displays the various features of the dataset.



Fig. 7. The correlation heatmap generated from the primary features of the dataset.

430 feature interpretability while optimizing computational efficiency. The systematic approach effectively balanced dataset 431 integrity with predictive utility, crucially enhancing model 432 generalizability through reduced overfitting risks from redundant 433 or anomalous data points.

435

B. Model training

436 In the development of deep learning models, the processes 437 of training and validation are essential for ensuring robust performance 438 and generalization on unseen data. These steps are 439 particularly vital in deep learning frameworks, as they help 440 optimize the integration of various algorithms to achieve peak

441 performance. For this study, we utilized the Z boson Dataset, 442 which contains 20,000 events, each characterized by 10 features. The dataset was partitioned into three distinct sub- 443 sets: 70% allocated for training, 10% for validation, and 20% 444 for testing. This stratified division ensures that the model is 445 trained and evaluated on non-overlapping data segments, pro- 446 viding a comprehensive assessment of its performance. Such 447 an approach not only mitigates potential biases arising from 448 imbalanced data distributions but also enhances the model's 449 ability to generalize across diverse data scenarios. By rig- 450 orously evaluating the model on separate validation and test 451 sets, we can better understand its behavior and ensure its reli- 452 ability in real-world applications.

454 1. Model Architecture and Training Configuration

455 This work proposes a hybrid neural network architecture
 456 that integrates deep representation learning and interpretability.
 457 The core design is realized through a multimodal feature fusion mechanism, an interpretable classification head,
 458 and an optimization strategy. Firstly, the model adopts a parallel dual-branch structure for heterogeneous feature extraction.
 459 The local feature capturing branch is based on a 1D temporal convolutional neural network to capture the local
 460 kinematic correlations of particle decay products. Hierarchical module stacking is employed for feature learning, with
 461 global average pooling layers used to compress spatiotemporal features while retaining key physical patterns. The deep
 462 residual learning branch constructs a four-stage residual network, with each stage consisting of two basic ResNet modules.
 463 The network extracts multi-scale features through incremental channel growth and layered down sampling, alleviating
 464 the vanishing gradient problem. The dual-branch outputs are concatenated through channel-wise fusion and fed
 465 into an interpretable classification head, optimizing both local and global features. The interpretable classification head
 466 uses a Kolmogorov-Arnold representation theorem-based interpretable network (KAN), replacing traditional activation
 467 functions with learnable B-spline basis functions. The decision boundary is explicitly expressed through the parametric
 468 combination of basis functions. The network structure is automatically configured according to the test input, ensuring
 469 that the input dimension strictly matches the fused feature dimension. The output layer utilizes temperature-scaled
 470 Softmax to balance classification confidence and the need for physical interpretability.

471 In terms of optimization strategy, the training process
 472 adopts a quadruple regularization scheme to improve the
 473 model's generalization ability. First, a dynamic learning
 474 rate decay strategy based on an exponential scheduler is
 475 used to enable rapid convergence in the early stages of
 476 training, followed by fine-tuning in later stages. Second, the AdamW optimizer is employed with weight decay (weight_decay=1×10⁻⁴) to prevent over-parameterization
 477 of the weight matrix. The initial learning rate is set to
 478 1×10⁻³ and is adapted through the exponential scheduler. Implicit regularization is introduced through batch normalization layers in the CNN/ResNet branches and a 0.2 probability Dropout layer, which synergistically enhance feature robustness. During training, cross-entropy is used as the loss function, as described by the formula:

$$500 L = -\frac{1}{N} \sum_{i=1}^N \sum_{c=1}^C y_{ic} \log(p_{ic}), \quad (11)$$

501 where p_{ic} represents the predicted probability that sample i
 502 belongs to class c . Furthermore, in terms of computational
 503 efficiency optimization, the batch size is set to 64 samples per
 504 batch, balancing GPU memory efficiency and gradient update
 505 stability. This architecture ensures high representational
 506 power of the deep model while meeting the interpretability
 507 requirements of particle physics experiments for model deci-

508 sion transparency, providing empirical evidence for hyperparameter optimization.

510 2. Performance Evaluation and Ablation Study

511 In this study, to comprehensively evaluate the performance
 512 of the proposed ensemble learning model, several standardized
 513 evaluation metrics were employed, including accuracy,
 514 precision, recall, and F1 score (Table 1). Accuracy is one
 515 of the most commonly used metrics for assessing the performance
 516 of classification models, representing the proportion
 517 of correctly classified samples out of the total number of samples.
 518 The basic formula for accuracy is given by:

$$519 \text{Accuracy} = \frac{\text{Number of Correct Predictions}}{\text{Total Number of Predictions}} \\ = \frac{TP + TN}{TP + TN + FP + FN}. \quad (12)$$

520 In this formula, TP (True Positive) refers to the number
 521 of samples correctly classified as the positive class; TN
 522 (True Negative) refers to the number of samples correctly
 523 classified as the negative class; FP (False Positive) refers to
 524 the number of samples incorrectly classified as the positive
 525 class; FN (False Negative) refers to the number of samples in-
 526 correctly classified as the negative class. Precision is a metric
 527 used to assess the proportion of true positive samples among
 528 all samples predicted as positive by the model. The funda-
 529 mental formula for precision is:

$$530 \text{Precision} = \frac{TP}{TP + FP}, \quad (13)$$

531 where TP (True Positive) refers to the number of samples
 532 correctly classified as the positive class; FP (False Positive)
 533 refers to the number of samples incorrectly classified as the
 534 positive class. Recall is a metric used to measure the pro-
 535 portion of actual positive samples correctly identified by the
 536 model. The basic formula for recall is:

$$537 \text{Recall} = \frac{TP}{TP + FN}, \quad (14)$$

538 where TP (True Positive) refers to the number of samples
 539 correctly classified as the positive class; FN (False Negative)
 540 refers to the number of samples incorrectly classified as the
 541 negative class. F1 score is the harmonic mean of precision
 542 and recall, combining both metrics to provide a more com-
 543 prehensive evaluation of model performance, particularly in
 544 cases of class imbalance. The formula for F1 score is:

$$545 \text{F1} = \frac{2 \times (\text{Precision} \times \text{Recall})}{\text{Precision} + \text{Recall}}. \quad (15)$$

546 The value of the F1 score ranges from 0 to 1, with higher
 547 values indicating a better balance between precision and
 548 recall. We selected several machine learning models (CNN,
 549 W-KNN, KD-KNN, KNN, XGB) from recent literature and
 550 online resources, with the aim of distinguishing between the

551 two decay processes of the Z boson. A comparative analysis
 552 was conducted to verify the effectiveness and superiority of
 553 our method. As shown in TABLE 1, our model demonstrates
 554 a significant improvement in accuracy compared to the other
 555 models.

TABLE 1. The accuracy, precision, recall, and F1 score of each classifier.

Model	Precision	Recall	F1 score	Accuracy	TT (sec)
W-KNN	69.20%	69.07%	69.01%	69.05%	0.060
KD-KNN	68.18%	67.93%	67.80%	67.90%	0.053
KNN	66.93%	66.87%	66.82%	66.85%	0.055
CNN	79.30%	72.06%	86.07%	75.98%	0.460
XGB	81.87%	81.45%	81.36%	81.42%	1.334
KAN-ResCNN	96.90%	92.70%	94.75%	94.73%	2.678

556
 557

TABLE 2. An ablation study of each module in KAN-ResCNN for the classification of Zmumu and Zee decay modes, where \checkmark denotes the addition of corresponding components or strategies.

ResNet	CNN	KAN	Precision	Recall	F1 score	Accuracy
\checkmark			90.04%	79.34%	84.35%	84.90%
	\checkmark		79.30%	72.06%	86.07%	75.98%
\checkmark	\checkmark		91.17%	84.50%	87.71%	87.85%
	\checkmark	\checkmark	88.78%	81.38%	84.92%	85.18%
\checkmark	\checkmark	\checkmark	96.90%	92.70%	94.75%	94.73%

558

559 To validate the effectiveness of the architectural design, we
 560 conducted an ablation study on the KAN-ResCNN model,
 561 with detailed results presented in TABLE 2. In this study,
 562 we explored the impact of various combinations of the
 563 ResNet, CNN, and KAN modules on the model’s perfor-
 564 mance. Specifically, we first retained the ResNet branch
 565 while removing both the CNN and KAN components to eval-
 566 uate the effectiveness of the deep residual network in feature
 567 extraction and overall model performance. Next, we only
 568 kept the CNN branch, removing ResNet and KAN, to as-
 569 sess the CNN’s capability in local feature extraction. Sub-
 570 sequently, we combined ResNet with KAN and CNN with
 571 KAN to examine the role of the KAN module in enhanc-
 572 ing model interpretability. By comparing the experimental
 573 results across different configurations, we found that the in-
 574 tegration of KAN with both ResNet and CNN significantly
 575 improved the model’s discriminative ability, especially in
 576 terms of Precision and Recall, thereby substantially enhanc-
 577 ing its overall performance. Consequently, the combination
 578 of ResNet, CNN, and KAN was identified as the optimal con-
 579 figuration, leading to significant improvements in both the ac-
 580 curacy and robustness of the model.

581 Fig. 8 presents three graphs that illustrate the model’s
 582 performance on different training metrics. The first plot,
 583 Loss Dynamics with Trendlines, has the training epochs
 584 (ranging from 1 to 150) on the x-axis and the loss values on
 585 the y-axis, which reflect the discrepancy between the model’s
 586 predictions and the true labels. Lower loss indicates bet-
 587 ter model performance. The blue solid line represents the
 588 training loss (Train Loss) across epochs, which initially starts
 589 high but decreases as the model learns, indicating the model’s

590 adaptation to the training data. The green solid line shows
 591 the validation loss (Val Loss), which ideally should decrease
 592 in parallel with the training loss. However, after epoch 35,
 593 the validation loss begins to increase, signaling overfitting.
 594 The dashed trendlines are quadratic polynomial fits to the
 595 loss curves, illustrating the overall trend. The blue dashed
 596 line corresponds to the training loss trend, while the green
 597 dashed line represents the validation loss trend. The R^2 value,
 598 which measures the fit between the trendline and the actual
 599 data (ranging from 0 to 1), is 0.96 in this case, indicating that
 600 the trendlines account for 96% of the variation in the data.
 601 The second plot, Accuracy Progression, also uses the train-
 602 ing epochs on the x-axis, with accuracy (ranging from 0 to
 603 1, where higher values indicate better performance) on the
 604 y-axis. The blue solid line represents the training accuracy
 605 (Train Acc), showing an increase in accuracy as the model
 606 learns. The green solid line represents the validation accuracy
 607 (Val Acc), which reflects the model’s performance on unseen
 608 data. The red dots (Best Acc) mark the highest validation ac-
 609 curacy achieved during training, indicating the model’s op-
 610 timal performance and corresponding epoch. The trendline
 611 further highlights the long-term trend in accuracy progres-
 612 sion. The third plot, F1-Score Evolution, shows the training
 613 epochs on the x-axis and the F1 score (ranging from 0 to 1) on
 614 the y-axis. The blue solid line represents the training F1 score
 615 (Train F1), which indicates the model’s ability to balance the
 616 identification of both positive and negative samples. The solid
 617 green line shows the validation F1 score (Val F1), with the
 618 trendline revealing the overall direction of the evolution of
 619 the F1 score throughout the training process. These visualiza-
 620 tions provide a comprehensive overview of the model’s per-
 621 formance across training and validation datasets, highlighting
 622 key metrics such as loss, accuracy, and F1 score, as well as
 623 trends indicating overfitting and optimal model performance.

625 V. SUMMARY

626 This study addresses the physical analysis re-
 627 quirements to classify Z boson decay modes and
 628 introduces the KAN-ResCNN model, which in-
 629 tegrates Kolmogorov-Arnold Networks (KAN),
 630 Residual Networks (ResNet), and
 631 Convolutional Neural Networks (CNN), achieving syn-
 632 ergistic optimization of high accuracy and physical inter-
 633 pretability. Large-scale experiments based on the publicly
 634 available CERN dataset demonstrate that the model achieves
 635 94.73% accuracy and 94.75% F1 score in the binary clas-
 636 sification tasks of $Z \rightarrow \mu^+ \mu^-$ and $Z \rightarrow e^+ e^-$, showing a
 637 significant improvement of 9.83% in accuracy over tradi-
 638 tional machine learning models (e.g., XGBoost, SVM) and
 639 mainstream deep learning models (e.g., pure ResNet, CNN).

640 The core innovations of the model are as follows. First,
 641 the heterogeneous feature extraction architecture of the dual
 642 branch uses the residual structure of the branch ResNet to
 643 model deep physical laws, combined with the local convo-
 644 lution operations of the branch CNN to capture detector-
 645 level features. This design enables complementary enhance-

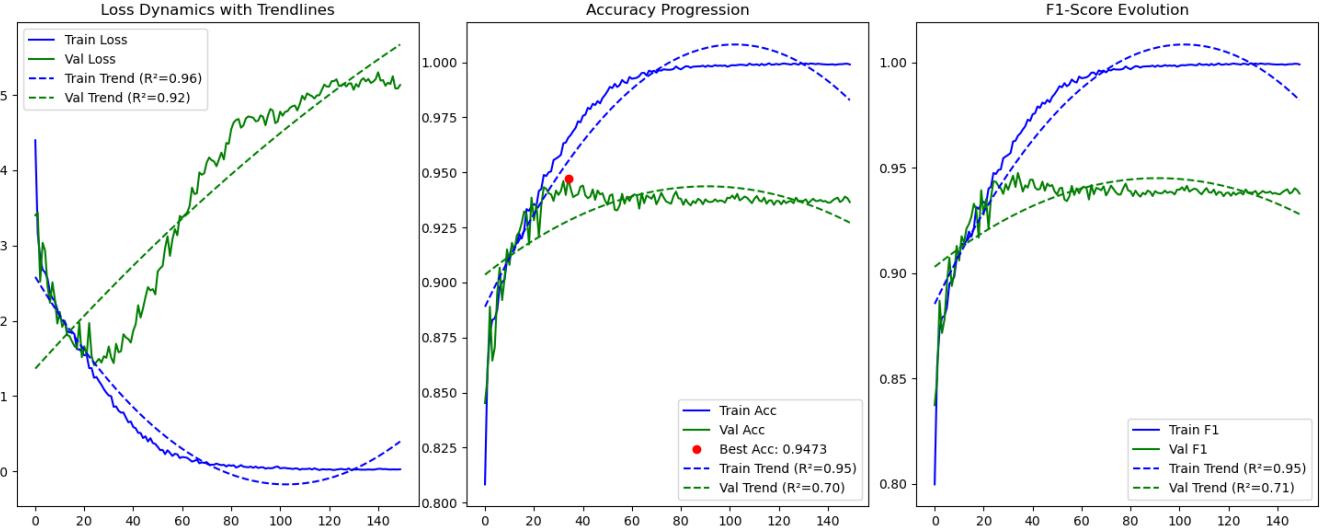


Fig. 8. The visualization metrics of the model on both the training and validation datasets.

646 ment of global physical laws and local feature representa-
 647 tions. Ablation experiments show that the dual-branch de-
 648 sign improves classification accuracy by 6.9% compared to a
 649 single-branch model. Second, the theory-driven KAN clas-
 650 sification decision leverages the interpretability advantage
 651 of Kolmogorov-Arnold representation theorem, transforming
 652 the black-box decision making of traditional neural networks
 653 into an explicit mapping based on spline functions and combi-
 654 nations of basis functions. The KAN module dynamically ad-
 655 justs the weights of the features through a grid update mech-
 656 anism, and its visualized decision path aligns closely with
 657 theoretical predictions of standard models, providing a new
 658 paradigm for the inversion of the physical mechanism. Fi-
 659 nally, data-driven, particle-physics-oriented optimization in-
 660 troduces innovative methods such as dynamic dimension fu-
 661 sion, residual block channel expansion, and physics-inspired
 662 regularization. Experiments show that the improved resid-
 663 ual block structure significantly improves the stability of the

664 gradient, while the B-spline-based L1-entropy regularization
 665 strategy effectively mitigates overfitting, resulting in a notice-
 666 able reduction in generalization error on the test set.

667 The results of this study provide new tools for intelligent
 668 analysis in high-energy physics experiments. The methodol-
 669 ogy is extendable to particle decay mode recognition tasks
 670 such as Higgs boson and top quark decay modes. Future
 671 work will focus on three aspects: extending the model to
 672 multi-decay channel classification, exploring dynamic model-
 673 ing of cascade decay processes; incorporating attention mech-
 674 anisms to strengthen the extraction of key physical features;
 675 and developing a KAN-based framework for physical law dis-
 676 covery, achieving a closed-loop analysis from data-driven to
 677 theory-driven approaches. This research offers technical sup-
 678 port for the deep integration of particle physics and artificial
 679 intelligence, with significant scientific value and application
 680 prospects.

681 [1] P. Langacker, M.X. Luo, Constraints on addi- 682 tional Z bosons. *Phys. Rev. D* **45**, 278-292 (1992).
 683 <https://link.aps.org/doi/10.1103/PhysRevD.45.278>

684 [2] V.A. Novikov, L.B. Okun, N.A. Rozanov, et al., The-
 685 ory of Z boson decays. *Rep. Prog. Phys.* **62**, 1275 (1999).
 686 <https://doi.org/10.1088/0034-4885/62/9/201>

687 [3] K, T, Experimental verification of the standard model of par-
 688 ticle physics. *Proc. Japan Acad., Ser. B* **97**, 211-235 (2021).
 689 <https://doi.org/10.2183/pjab.97.013>

690 [4] S. Antusch, J.P. Baumann, E. Fernández-Martínez, Non-
 691 standard neutrino interactions with matter from physics be-
 692 yond the Standard Model. *Nucl. Phys. B* **810**, 369-388 (2009).
 693 <https://doi.org/10.1016/j.nuclphysb.2008.11.018>

694 [5] P. Collier, The technical challenges of the Large Hadron Col-
 695 lider. *Philos. Trans. R. Soc. A Math. Phys. Eng. Sci.* **373**,
 696 20140044 (2015). <https://doi.org/10.2307/24506083>

687 [6] V. Olshevsky, Y. Olshevsky, A. Lalti, et al., Automated clas-
 688 sification of plasma regions using 3D particle energy distri-
 689 butions. *J. Geophys. Res. Space Physics* **126**, 2021JA029620
 690 (2021). <https://doi.org/10.1029/2021JA029620>

701 [7] E. De Bézenac, A. Pajot, P. Gallinari, Deep learning for
 702 physical processes: incorporating prior scientific knowl-
 703 edge. *J. Stat. Mech. Theor. Exp.* **2019**, 124009 (2019).
 704 <https://dx.doi.org/10.1088/1742-5468/ab3195>

705 [8] Z.M. Liu, Y.X. Wang, S. Vaidya, et al., Kan: Kolmogorov-
 706 arnold networks. *arXiv preprint arXiv:2404.19756* (2024).
 707 <https://doi.org/10.48550/arXiv.2404.19756>

708 [9] M. Kubu, P. Bour, CNN with residual learning extensions in
 709 neutrino high energy physics. *J. Phys. Conf. Ser.* **1730**, 012133
 710 (2021). <https://doi.org/10.1088/1742-6596/1730/1/012133>

711 [10] S.Y. Chen, T. Wei, C, Zhang, et al., Quantum con-
 712 volutional neural networks for high energy physics

713 data analysis. *Phys. Rev. Res.* **4**, 013231 (2022). <https://doi.org/10.1103/PhysRevResearch.4.013231>

714

715 [11] P.J. Sadowski, W. Daniel, P. Baldi, Searching for Higgs Bo-
716 son Decay Modes with Deep Learning. In: Z. Ghahramani, M.
717 Welling, C. Cortes, et al., (eds) *Advances in Neural Information
718 Processing Systems*, vol 27. (Curran Associates, Inc., 2014).
719 <https://proceedings.neurips.cc/paper-files/paper/2014>

720 [12] N. Ranasinghe, Y. Xia, S. Seneviratne, et al., Ginn-kan: In-
721 terpretability pipelining with applications in physics informed
722 neural networks. *arXiv preprint arXiv:2408.14780* (2024).
723 <https://doi.org/10.48550/arXiv.2408.14780>

724 [13] C.D. Sag, O. Sahin, Predicting Z Boson Decay Modes: Evalu-
725 ating the Performance of Machine Learning and Deep Learning
726 Techniques in Particle Physics. *IEEE (UBMK) 17-22* (2023).
727 <https://doi.org/10.1109/UBMK59864.2023.10286786>

728 [14] S.D. Lane, H.S. Lee, I.M. Lewis, Multi-photon decays of the
729 Higgs boson at the LHC. *arXiv preprint arXiv:2305.00013*
730 (2023). <https://doi.org/10.48550/arXiv.2305.00013>

731 [15] D.X. Wu, Y.S. Wang, S.T. Xia, et al., Skip connections mat-
732 ter: On the transferability of adversarial examples gener-
733 ated with resnets. *arXiv preprint arXiv:2002.05990* (2020).
734 <https://doi.org/10.48550/arXiv.2002.05990>

735 [16] A. Celik, Exploring hidden signal: Fine-tuning ResNet-50 for
736 dark matter detection. *Comput. Phys. Commun.* **305**, 109348
737 (2024). <https://doi.org/10.1016/j.cpc.2024.109348>

738 [17] K. Bobzin, H. Heinemann, SR Dokhanchi, et al., Replication
739 of Particle Trajectories in the Plasma Jet with Two Consecutive
740 Residual Neural Networks. *J. Therm. Spray Technol.* **32**, 1447-
741 1464 (2023). <https://doi.org/10.1007/s11666-023-01533-1>

742 [18] W. Ling, K.J. Pan, Z.Y. Ren, et al., One-dimensional mag-
743 netotelluric parallel inversion using a ResNet1D-8 resid-
744 ual neural network. *Comput. Geosci.* **180**, 105454 (2023)
745 <https://doi.org/10.1016/j.cageo.2023.105454>

746 [19] M. Jugin, M.S. Madhulika, G.D. Divya, et al., Fea-
747 ture extraction using convolution neural networks (CNN)
748 and deep learning. *IEEE (RTEICT) 2319-2323* (2018).
749 <https://doi.org/10.1109/RTEICT42901.2018.9012507>

750 [20] R.C. Yu, S. Wu, J. Gui, Residual kolmogorov-arnold network
751 for enhanced deep learning. *arXiv preprint arXiv:2410.05500*
752 (2024). <https://doi.org/10.48550/arXiv.2410.05500>

753 [21] H. Ide, T. Kurita, Improvement of learning for CNN with ReLU
754 activation by sparse regularization. In: *2017 International
755 Joint Conference on Neural Networks (IJCNN)*, pp. 2684-2691
756 (2017). <https://doi.org/10.1109/IJCNN.2017.7966185>

757 [22] H. Gholamalinezhad, H. Khosravi, Pooling methods in deep
758 neural networks, a review. *arXiv preprint arXiv:2009.07485*
759 (2020). <https://doi.org/10.48550/arXiv.2009.07485>

760 [23] R.U. Khan, X. Zhang, R. Kumar, et al., Evaluating the
761 performance of resnet model based on image recognition.
762 Proc. In: *Proceedings of the 2018 International Conference
763 on Computing and Artificial Intelligence*, pp. 86-90 (2018).
764 <https://doi.org/10.1145/3194452.3194461>

765 [24] Z.T. Huang, L.Y. Wang, Q.Y. Ge, et al., An intelli-
766 gent fault diagnosis method for CNN-SVM circuit breaker
767 based on quantum particle swarm optimization. *J. Phys.
768 Conf Ser.* **2113**, 012047 (2021). [https://doi.org/10.1088/1742-
770 6596/2113/1/012047](https://doi.org/10.1088/1742-
769 6596/2113/1/012047)

771 [25] S.H. Lim, M. Nojiri, Morphology for jet clas-
772 sification. *Phys. Rev. D* **105**, 014004 (2022).
773 <https://doi.org/10.1103/PhysRevD.105.014004>

774 [26] J.Y. Araz, M. Spannowsky, Combine and conquer:
775 event reconstruction with Bayesian ensemble neural
776 networks. *J. High Energy Phys.* **2021**, 1-23 (2021).
777 [https://doi.org/10.1007/JHEP04\(2021\)296](https://doi.org/10.1007/JHEP04(2021)296)

778 [27] S. Thaïs, P. Calafiura, G. Grigoriou, et al., Graph Neu-
779 ral Networks in Particle Physics: Implementations, Inno-
780 vations, and Challenges. *arXiv preprint arXiv:2203.12852*
781 (2022). <https://doi.org/10.48550/arXiv.2203.12852>

782 [28] C. Shimmin, Particle convolution for high energy
783 physics. *arXiv preprint arXiv:2107.02908* (2021).
784 <https://doi.org/10.48550/arXiv.2107.02908>

785 [29] A. Zafar, M. Aamir, N. Mohd Nawi, et al., A comparison of
786 pooling methods for convolutional neural networks. *Appl. Sci.*
787 **12**, 8643 (2022). <https://doi.org/10.3390/app12178643>

788 [30] M. Révayová, C. Török, Piecewise approximation and
789 neural networks. *Kybernetika* **43**, 547-559 (2007).
790 <http://www.kybernetika.cz/content/2007/4/547>

791 [31] J.E. Dayhoff, J.M. DeLeo, Artificial neural networks:
792 opening the black box. *Cancer* **S8**, 1615-1635 (2001).
793 <https://doi.org/10.1002/1097-0142>